78-FM-30
Vol. VIII

JSC-14266

OPS MCC Ground Navigation Program
Level C
Orbit Determination Processing

Formulation Requirements

Data Editing Processor (DEP)

Mission Planning and Analysis Division

September 1979

National Aeronautics and
Space Administration

Lyndon B. Johnson Space Center
Houston, Texas

78-FM-30
Vol. VIII

JSC-14266

SHUTTLE PROGRAM

OPS MCC
GROUND NAVIGATION PROGRAM

LEVEL C SOFTWARE REQUIREMENTS
ORBIT DETERMINATION PROCESSING FORMULATIONS DOCUMENT

DATA EDITING PROCESSOR (DEP)

By M. D. Zuteck and L. G. Putnam
TRW Defense and Space Systems Group

JSC Task Monitor: J. B. Williamson, Mathematical Physics Branch

Approved: _____
Wilbur R. Wollenhaupt, Head
Ground Navigation Section

Approved: _____
Emil R. Schiesser, Chief
Mathematical Physics Branch

Approved: _____
Ronald L. Berry, Chief
Mission Planning and Analysis Division

Mission Planning and Analysis Division

National Aeronautics and Space Administration

Lyndon B. Johnson Space Center

Houston, Texas

September 1979

PREFACE

The Mathematical Physics Branch/Mission Planning and Analysis Division has the responsibility to provide the functional ground navigation software formulation requirements for the Mission Control Center (MCC) low-speed-processing phases during Operations Project Shuttle (OPS).

The ground navigation software formulation requirements are logically organized into volumes. This organization is presented in the accompanying table. The material in each volume presents the level C formulation requirements of the processors and modules required to process low-speed-tracking data and perform orbit determination and other related navigation computations. Each volume describes the formulation requirements of the identified processor or module specified in the OPS MCC Ground Navigation Program Level B Software document (ref. 1). The inputs and outputs required to accomplish the functions described are specified. Flow charts defining the sequence of mathematical operations and display and control processing required to satisfy the described functions are included in the document where appropriate.

OPS MCC GROUND NAVIGATION PROGRAM LEVEL C SOFTWARE REQUIREMENTS

ORBIT DETERMINATION PROCESSING FORMULATIONS DOCUMENT

Volume I	Introduction and Overview
Volume II	Low-Speed Input Processor (LSIP)
Volume III	Bias Correction Processor (BCP)
Volume IV	Data File Control Processor (DFCP)
Volume V	Orbit Determination Executive (ODE)
Volume VI	Convergence Processor (CP)
Volume VII	Differential Correction Module (DCM)
Volume VIII	Data Editing Processor (DEP)
Volume IX	Covariance Matrix Processor (CMP)
Volume X	State Transition Matrix Module (STMM)
Volume XI	Observation Computation Module (OCM)
Volume XII	Measurement Partial Derivative Module (MPDM)
Volume XIII	Residual Computation Processor (RCP)
Volume XIV	Display Processor

VOLUME VIII

DATA EDITING PROCESSOR (DEP)

CONTENTS

1.0 CORRELATION TO LEVEL B

This document presents the level C software requirements that satisfy the automatic editor portion of the level B software formulation requirements specified for the data editing processor in sections 5.11, 6.0 (fig. 6-10), 7.2.10, and 8.3.10 of JSC internal note 77-FM-57 (ref. 1).

These level C software requirements also satisfy the level B software display and control requirements specified for data editing in sections 6.4.9(c) and (d) of JSC internal note 77-FM-56 (ref. 2). It should be noted that the manual (N_1N_2) edit function is now documented in the display processor document.

2.0 GENERAL DESCRIPTION

2.1 BASIC CAPABILITIES AND LIMITATIONS

The automatic editor described herein is designed to detect and edit data points whose residuals lie outside the general pattern of the data residuals by a significant amount. Its operation is based on a comparison of the local correlation of small groups of points with the corresponding correlation for the residual population as a whole. Because it employs small local groups, it can detect outliers that could be masked by the systematic structure of the residuals if another type of editor was used.

This editor has the capability to detect and edit isolated or paired outliers and to edit the clearly inappropriate members of noisy groups. It can also tolerate the wide range of residual values, the data gaps, and the discontinuities that sometimes occur in typical residual populations without significant degradation of its editing performance. However, it does not attempt to automatically resolve all possible editing situations, because it is recognized that there exist complex and subtle situations where even the full capabilities of an experienced human operator are heavily taxed. Such situations are considered the proper province of the experienced and responsible human operator; therefore, the editor has been designed to alert the operator in such cases.

A number of user-defined variables exist by which the editing operation can be controlled and tuned. These include (1) an editing minimum for each data type, below which editing will never occur; (2) an editing maximum for each data type that sets the limit for an initial gross edit and thereby also provides a limit by which to judge that the overall residual population has become unacceptably large; and (3) certain other variables that control the way in which the automatically computed edit criterion is set and when the operator is to be alerted. These variables provide powerful user control over the basic aggressiveness or conservatism of the automatic editing, so that it can be tuned to specific needs as they arise.

2.2 OVERVIEW OF THE EDITOR CONCEPTS AND OPERATIONS

The operation of the present editor is based on the use of divided first and second differences of the residual data. The central idea, expressed in its simplest form, is that both the deterministic structure of the data and the usual random measurement noise will lead to a rather well-bounded range of values for the divided differences, whereas outliers will generally cause values well outside that range. That characteristic of divided differences makes them a potentially powerful editing tool, but the practical realization of an editing scheme based on that foundation requires a number of additional concepts, the net sum of which largely dictates the structure of the editor.

It is essential to recognize that the size of the bounded range of difference values due to deterministic structure and random noise is going to change dramatically from one residual population to another. Partly, this is a result of dealing with several different data types, such as range, Doppler, and angles, that have distinctly different typical magnitudes. But the magnitude and structure of the residuals are also very strongly influenced by how accurate the state vector used to generate the residuals happens to be. The result of this is that first and second difference editing bounds appropriate for one problem can be wildly inappropriate for another; and, therefore, these bounds have to be found for each problem based on its particular data. To achieve this, the absolute magnitudes of the differences are examined in order to find the break that will exist if outliers are present in the data. When such a break is detected, it provides a basis for setting the edit level for the data of that particular problem, because it provides a method for finding a difference value that corresponds to the upper limit of the range resulting from deterministic structure and normal measurement noise.

The actual edit level setting operation is not best done on the simple first and second differences, because an errant point will also corrupt the difference values for its neighbors. In addition, discontinuities and sharp changes in slope sometimes exist within the data, and they will cause large values for points that are perfectly valid members of the string of points to either side of the abrupt change. Therefore, a concept called best difference is introduced, where each point is allowed its best level of correlation with its neighbors; i.e., its smallest differences in the absolute value sense. A residual that "fits in" with its neighbors to one side or the other is going to have a small best difference. Only true outliers, which do not fit in well with their neighbors, will have large best differences. Therefore, the editing operations are, in fact, performed using these best difference values.

Another fact that must be accounted for when using divided differences for editing is that in an area with a high density of bad points, interspersed good points can possess very large best difference values purely because they are surrounded by errant neighbors. This makes them indistinguishable from bad points on the first edit pass. Even if a point has only one errant neighbor, that will increase its statistical chance of exceeding the edit limit, because it then has only one chance remaining to obtain acceptable difference values. To minimize the loss of good data due to these facts, the editor design provides for a check of any group of consecutively edited points. This check computes

new differences against accepted points, so that any points that fit in with the accepted data can be accepted, even though originally corrupted by bad points.

The above concepts and operations outline the basic editor operation. The detailed design also possesses the tests that provide the user-defined minimum and maximum edit limits for each data type and the various checks required to alert the user when situations requiring his attention are encountered. The detailed requirements for the editor realization follow.

3.0 EDITOR DETAILED FORMULATION REQUIREMENTS

3.1 GENERAL REQUIREMENTS

3.1.1 Data Processing and Composition

The editor shall process one batch of data per call. The various data types that exist in the batch are to be treated sequentially, one data type at a time, until all have been considered, at which time the editor shall return control to the calling program. Within a data type, each data point is the residual of a valid observable, with an associated time tag. It is assumed that these data are provided in time sequence. Time tags corresponding to missing measurement values shall be eliminated for each data type before it is edited, so that only residuals and times based on actual measured observables are used in the editing process.

3.1.2 Time Sequence of Major Editor Functions

There are five major editor functions that shall be performed sequentially in the following order: (1) gross preedit (2) D_1 edit, (3) D_2 edit, (4) group reedit, and (5) edit state tests.

3.1.3 Editor Flags

The editor shall possess flags for the following indications: (1) user review of data required, and (2) edit state has been altered during current pass. In order to satisfy item 2, and to monitor intermediate states during editing, there must also be a set of working edit flags, assigned one per data point that exist in addition to the regular edit flags carried with the data. A value of zero for a working edit flag shall indicate that the corresponding data point has not been edited. A value of one shall indicate that the data point has been edited. All working edit flags shall be set to zero before the Gross Preedit Logic (sec. 3.1.4) is performed. Note that in the following sections, unedited data points are defined to be those data points whose corresponding working edit flags are zero. The requirements for setting all the above flags are set forth in the appropriate sections of these requirements.

3.1.4 Gross Preedit

Before beginning the D_1, D_2 edit sequence, the residuals of the current data type shall be compared against the user-specified upper limit Rmax for that data type. Residuals whose absolute magnitude exceeds that limit shall have their working edit flags set to one and shall be deleted from the sequence of residuals passed on to D_1 editing. If the number of residuals so edited exceeds N·ELIM, where N is the initial number of residuals of that data type, the stop flag shall be set to one, and the edit termination procedure of section 3.7 shall be executed.

3.2 D_1 EDIT FUNCTION

After completion of preedit of section 3.1.4, there shall be at least MIN_1 data points with working edit flags clear in order to perform the D_1 edit, where MIN_1 is a specifiable integer constant. If this condition is not met, the stop flag shall be set to one, and the edit termination procedure of section 3.7 shall be executed.

The first editing operation on the data shall be based on the divided first differences of the residual data.

3.2.1 Divided First Difference Calculation (D_1)

The divided first difference shall be calculated for each interval between neighboring valid unedited data points according to the equation

$$D_1(i) = \frac{R(i+1) - R(i)}{t_{i+1} - t_i} \; ; \; i = 1,2,3,\ldots,N-1$$

where $R(i)$ is the ith data residual

t_i is the time tag of the ith data residual

N is the number of residuals of the current data type remaining after completing the gross preedit of section 3.1.4.

3.2.2 Best Divided First Difference Calculation (D_{1b})

A best divided first difference shall be found for each valid unedited data point according to the following equations:

$$D_{1b}(i) = \min\left(|D_1(i)|, |D_1(i-1)|\right) \; ; \; i = 2,3,\ldots,N-1$$

$$D_{1b}(1) = |D_1(1)|$$

$$D_{1b}(N) = |D_1(N-1)|$$

where min is the minimum value selection function

The resulting D_{1b} values shall be tagged to indicate the residual to which they correspond.

3.2.3 D_1 Edit Limit (D_1LIM)

The D_1 edit limit, D_1LIM, shall be chosen from the D_{1b} values based on the following rules:

$$D_1\text{LIM} = \text{Max}\ (D_{1b})$$

unless for $D_{1b}(n) \geq D_{1b}(\text{ref})$

either $D_{1b}(g) > (1+\text{DELTA1})\ D_{1b}(n)$

or $D_{1b}(g) > K1 \cdot D_{1b}(\text{ref})$

in which case $D_1\text{LIM} = D_{1b}(n)$ where the smallest value of $D_{1b}(n)$ is to be chosen if more than one satisfy the inequalities. However, if the D_1LIM so chosen is less than the user specified minimum divided by Δt; i.e., if

$$D_1\text{LIM} < \frac{R_{MIN(i)}}{\Delta t}$$

then $$D_1\text{LIM} = \frac{R_{MIN(i)}}{\Delta t}$$

where Max ( ) is the maximum value function

$D_{1b}(\text{ref})$ is the largest D_{1b} value that is not larger than $S_1 \cdot N$ of the D_{1b} values

S_1 is a user-defined limit $0 \leq S_1 < 1.0$

DELTA1 is a user-defined limit $0 < \text{DELTA1}$

K_1 is a user-defined limit $1 < K_1$

$D_{1b}(g)$ is the smallest D_{1b} value that is greater than $D_{1b}(n)$

$D_{1b}(n)$ is the current choice of D_{1b} for applying the inequalities

$R_{MIN(i)}$ is a user-defined minimum for each measurement type

i is the measurement type of the data currently being edited

Δt is the nominal time between successive data points and is computed as follows:

$$\Delta t = \frac{t_n - t_1}{n - 1}$$

3.2.3.1 Implementation of D_1 Edit Limit Determination Function

The formulation of the D_1 edit limit determination function in section 3.2.3 defines an algorithm in which a D_1 edit limit value is selected from a subset of the D_{1b} values for the data points in the batch. The subset of D_{1b} values from which the limit is selected is defined as those points whose D_{1b} is greater than D_{1b}(ref). Determination of the points in this subset may be facilitated by first sorting the D_{1b} in ascending order. The subset defined above will then be those D_{1b} values whose index is larger than the index of D_{1b}(ref). The index for D_{1b}(ref) is defined as the integer part of S1·N. Note that when the D_{1b} are sorted, some mechanism must be devised through which the sorted D_{1b} may be related to their respective time tags.

Sorting the D_{1b} will also facilitate the evaluation of the D_1 edit limit criterion for the points within the subset. The smallest point within the subset; i.e. ref+1, will be evaluated first. Thus D_{1b}(n) will initially be D_{1b}(ref+1) and D_{1b}(g) will be D_{1b}(ref+2). If D_{1b}(n) fails the criterion, the next point, n+1, will be evaluated. Each successive point will be evaluated until either one of the points passes the criterion or all the points in the subset have been evaluated. If a point passes the criterion, D_1LIM will be set to the D_{1b} for that point. If no points pass the criterion, D_1LIM will be set to D_{1b}(N).

3.2.4 D_1 Edit

The D_1 edit shall be performed by setting the working edit flag for each point with $D_{1b} > D_1$LIM to one.

3.3 D_2 EDIT FUNCTION

The second editing operation on the data shall be based on the divided second differences of the data.

After completion of D_1 edit of section 3.2.4, there shall be at least MIN_2 data points with working edit flags clear in order to perform D_2 edit, where

MIN_2 is a specifiable integer constant. If this condition is not met, D_2 edit shall be skipped and the REEDIT logic of section 3.4 shall be performed.

3.3.1 Divided Second Difference Calculation (D_2)

The divided second difference shall be calculated for each valid unedited data point that possesses a valid unedited neighbor (not necessarily adjacent) to both sides, according to the equation:

$$D_2(j) = \frac{D_1(j) - D_1(j-1)}{t_{j+1} - t_{j-1}} \qquad j=2,3,\ldots,J-1$$

where $D_1(j)$ is the jth first difference, as defined previously, except that no edited points are allowed in the $D_1(j)$ calculations, and the new index j is used simply to clarify that the edited points were skipped in the numbering

t_j is the jth time tag

J is the number of unedited points after D_1 editing.

3.3.2 Best Divided Second Difference Calculation (D_{2b})

The best divided second difference shall be found for each valid unedited data point according to the equations

$$D_{2b}(j) = \min\left(|D_2(j-1)|, |D_2(j)|, |D_2(j+1)|\right) \quad ; \quad j=3,\ldots,J-2$$

$$D_{2b}(1) = |D_2(2)|$$

$$D_{2b}(2) = \min\left(|D_2(2)|, |D_2(3)|\right)$$

$$D_{2b}(J-1) = \min\left(|D_2(J-2)|, |D_2(J-1)|\right)$$

$$D_{2b}(J) = |D_2(J-1)|$$

where min is the minimum value selection function.

The resulting D_{2b} values shall be tagged to indicate the residual to which they correspond.

3.3.3 D_2 Edit Limit (D_2LIM)

The D_2 edit limit, D_2LIM, shall be chosen from the D_{2b} values based on the following rules:

$D_2LIM = Max\ (D_{2b})$

unless, for $D_{2b}(n) \geq D_{2b}(ref)$

either $D_{2b}(g) > (1+DELTA2)\ D_{2b}(n)$

or $D_{2b}(g) > K2.D_{2b}(ref)$

in which case $D_2LIM = D_{2b}(n)$ where the smallest value of $D_{2b}(n)$ is to be chosen if more than one satisfy the inequalities. However, if the D_2LIM so chosen is less than the user specified minimum divided by Δt^2; i.e., if

$$D_2LIM < \frac{R_{MIN(i)}}{\Delta t^2}$$

then

$$D_2LIM = \frac{R_{MIN(i)}}{\Delta t^2}$$

where max ( ) is the maximum value function

$D_{2b}(ref)$ is the largest D_{2b} value that is not larger than $S_2 \cdot J$ of the D_{2b} values

S2 is a user-defined limit $0 \leq S_2 < 1.0$
DELTA2 is a user-defined limit $0 < DELTA2$

K2 is a user-defined limit $1 < K2$

$D_{2b}(g)$ is the smallest D_{2b} value that is greater than $D_{2b}(n)$

$D_{2b}(n)$ is the current choice of D_{2b} for applying the inequalities

$R_{MIN}(i)$ is a user-defined minimum for each measurement type.

i is the measurement type of the data currently being edited

Δt is as computed for D_1 edit

3.3.3.1 Implementation of D_2 Edit Limit Determination Function

The algorithm for determining the D_2 edit limit defined in section 3.3.3 is the same as the algorithm for determining the D_1 edit limit defined in section 3.2.3; thus, what was presented in section 3.2.3.1 about implementation of the D_1 edit limit determination function is applicable to the implementation of the

D_2 edit limit determination function, except any reference made to D_1 or D_{1b} should be replaced with D_2 or D_{2b}.

3.3.4 D_2 Edit

The D_2 edit shall be performed by setting the working edit flag for each point with $D_{2b} > D_2LIM$ to one.

3.4 REEDIT FUNCTION

Whenever a group of two or more consecutively edited points exist after the D_1, D_2 edit sequence, the reedit logic of sections 3.4.1 and 3.4.2 shall be executed. Note that points edited by preedit are not to be considered for reediting.

3.4.1 Recompute D_{1b} and D_{2b} for Edited Groups

For the points of any group of two or more consecutively edited points, new values of D_{1b} and D_{2b} are to be calculated, using the unedited point(s) outside the group in the calculation, except that if the D_2 edit sequence is skipped, new D_{2b} values need not be calculated.

The equations to be used for D_{1b} recomputation are

$$D_{1e} = \frac{R_c - R_e}{t_c - t_e}$$

$$D_{1\ell} = \frac{R_\ell - R_c}{t_\ell - t_c}$$

$$D_{1b} = \text{Min}\,(|D_{1e}|, |D_{1\ell}|)$$

where R_c is the residual at the current recomputation point

t_c is the time tag at the current recomputation point

R_e is the residual at the first unedited point earlier than the edited group

t_e is the time tag of Re

D_{1e} is the divided first difference between the current recomputation point and the first unedited earlier point

R_ℓ is the residual at the first unedited point later than the edited group

t_ℓ is the time tag of R_e

$D_{1\ell}$ is the divided first difference between the current recomputation point and the first unedited later point

D_{1b} is the recomputed D_{1b} value at the current point

Note: If the edited group occurs at either the beginning or end of the data batch, $D_{1e}(c)$ or $D_{1\ell}(c)$, respectively, cannot be computed because the required earlier or later unedited point does not exist. In that case, D_{1b} is just the absolute value of the divided first difference, which can be calculated, using the available unedited point.

The equations to be used for D_{2b} recomputation are

$$D_2 = \frac{D_{1\ell} - D_{1e}}{t_\ell - t_e}$$

$$D_{2e} = \frac{D_{1e} - D_{1ee}}{t_c - t_{ee}}$$

$$D_{1ee} = \frac{R_e - R_{ee}}{t_e - t_{ee}}$$

$$D_{2\ell} = \frac{D_{1\ell\ell} - D_{1\ell}}{t_{\ell\ell} - t_c}$$

$$D_{1\ell\ell} = \frac{R_{\ell\ell} - R_\ell}{t_{\ell\ell} - t_\ell}$$

$$D_{2b} = \text{Min}\ (|D_{2e}|, |D_2|, |D_{2\ell}|)$$

where D_2 is a recomputed D_2 value at the current point, using the nearest unedited point to each side

D_{2e} is a recomputed D_2 value for the current point, using the two nearest earlier unedited points

D_{1ee} is the divided first difference between the two nearest unedited earlier points

R_{ee} is the residual at the second nearest earlier unedited point

t_{ee} is the time tag at the second nearest earlier unedited point

$D_{2\ell}$ is the recomputed D_2 value for the current point, using the two nearest unedited later points

$D_{1\ell\ell}$ is the divided first difference between the two nearest unedited later points

$R_{\ell\ell}$ is the residual at the second nearest later unedited point

$t_{\ell\ell}$ is the time tag at the second nearest later unedited point

D_{2b} is now the recomputed D_{2b} value at the current point

Note: If the edited group occurs at either the beginning or end of the data batch, D_2 and either D_{2e} or $D_{2\ell}$, respectively, cannot be computed because the required earlier or later unedited points do not exist. In that case, the endpoint D_{2b} is just the absolute value of the one D_2 which can be recomputed, using unedited points, while the D_{2b} for the point one in from the endpoint is the absolute value of the minimum of the two D_2's which can be recomputed, using unedited points.

3.4.2 Reedit

A recomputed D_{1b} value must satisfy the inequality $D_{1b} < D_1LIM$ in order for a point to have its associated working edit flag reset to zero.

If D_2 edit was executed, then a recomputed D_{2b} value must satisfy the inequality $D_{2b} < D_2LIM$ in order for a point to have its associated working edit flag reset to the cleared state.

If both D_1 and D_2 edit were performed, then both inequalities must be satisfied in order for a point to have its associated working edit flag reset to the cleared state.

3.5 EDIT STATE TESTS

After completion of the D_1, D_2 edit sequence, four edit state tests are to be applied to the current working edit flags.

3.5.1 Data Discontinuity Test

A test shall be performed to determine whether $|D_1| > D \cdot D_1LIM$, with neither endpoint of that D_1 computation interval edited, where D is a constant $D > 1$.

This test is defined on the set of all intervals between neighboring unedited points. When the test condition is detected, the stop flag shall be set to one.

3.5.2 Total Edit Percentage Test

A test shall be performed to determine whether the total number of points edited exceeds a given percentage of the initial number of points (that percentage being set by the edit limit parameter ELIM). When that limit is exceeded, the stop flag shall be set to one.

3.5.3 Large Edited Group Test

A test shall be performed to determine whether a group of G or more consecutive edited points exists, where G is a constant, G > 1 and G integer. If it does, the the stop flag shall be set to one.

3.5.4 Edit Alter Test

Set the edit alter flag to one if the current pass through the DEP has caused a change in the batch edit status.

3.6 REGULAR EDIT FLAG ALTERATION

After all data types for the current batch have been processed, the last operation shall be to perform the edit alter test, and to set the regular edit flags equal to the working edit flags. This operation is not to be performed if the tests on the current batch have caused the stop flag to be set to one.

3.7 EDITING TERMINATION AND RETURN

If a condition is encountered that causes the stop flag to be set, all regular edit flags for all data types in the current batch shall be unchanged from input and editing shall terminate with a stop error condition return.

```
-------------------------------------------------------------------------
! When editing proceeds in a normal fashion and no stop condition is encountered,!
! the editor shall set the batch edited flag in the batch header record upon com-!
! pletion of the last data type, and then provide a normal return.              !
-------------------------------------------------------------------------
```

4.0 INPUTS

See ODE to DEP interface table for a description of inputs.

5.0 OUTPUTS

See DEP to ODE interface table for a description of outputs.

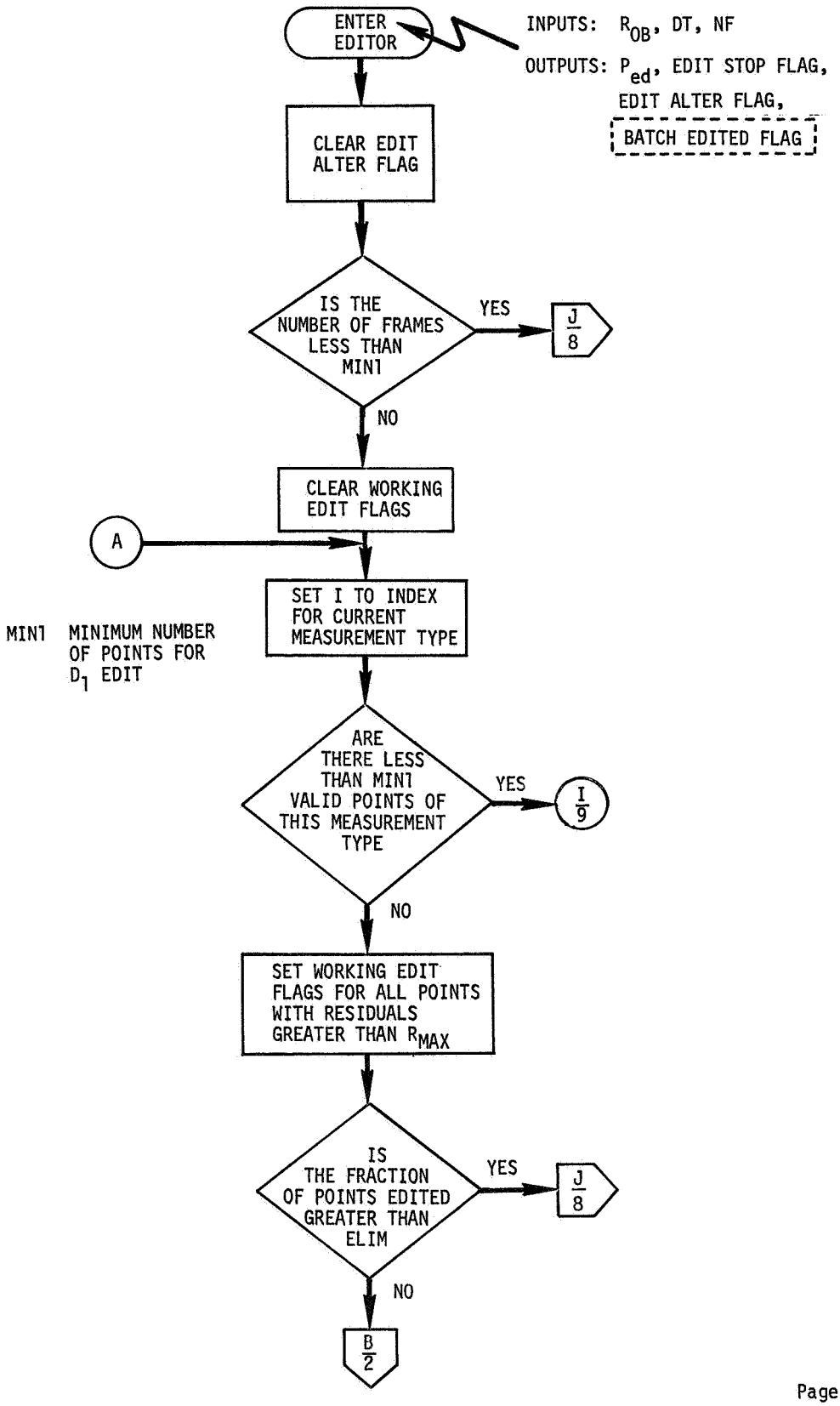
6.0 REFERENCES

1. Bergman, O. R.; McMahon, J. J.; and Newman, C. M.; and Williamson, J. B.: OPS MCC Ground Navigation Program, Level B Software Preliminary Orbit Determination Processing Formulation Requirements. JSC IN 77-FM-57, Oct. 1977.

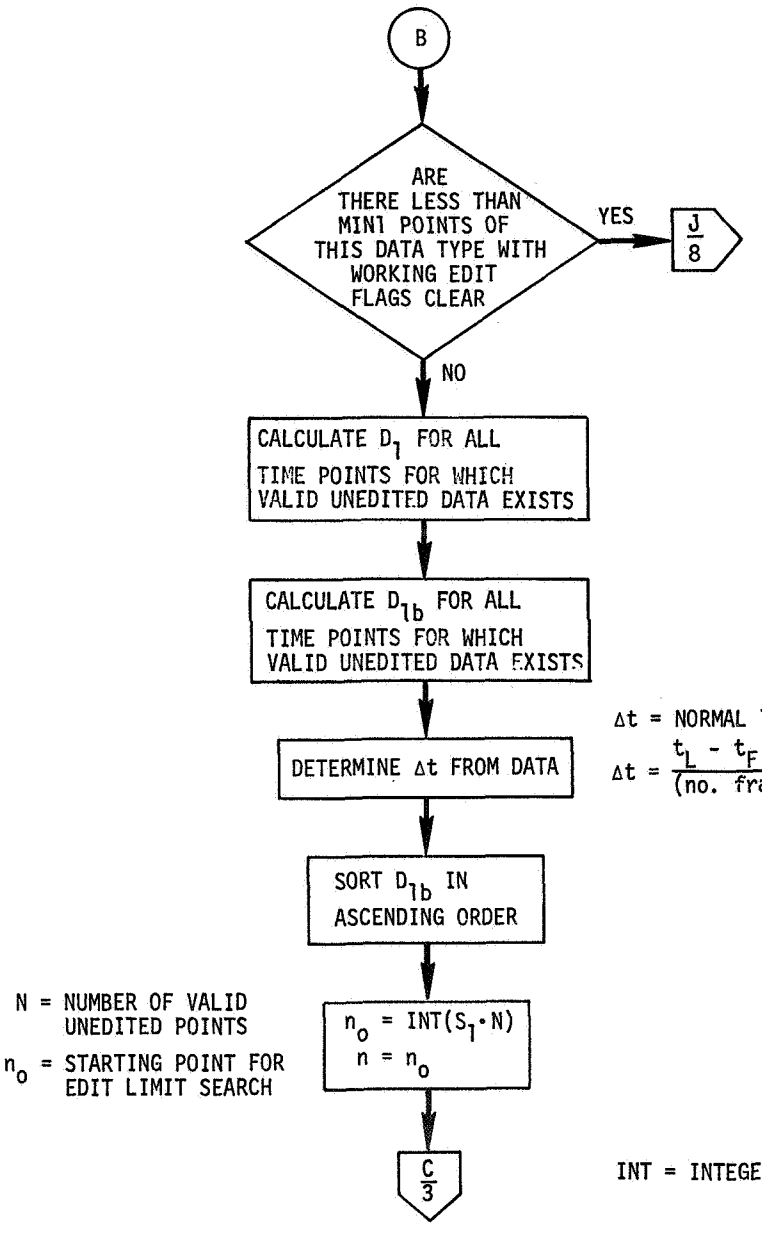
2. Osburn, R. K.; and Wollenhaupt, W. R.: OPS MCC Ground Navigation Program, Level B Software Display and Control Requirements. JSC IN 77-FM-56, Oct. 1977.

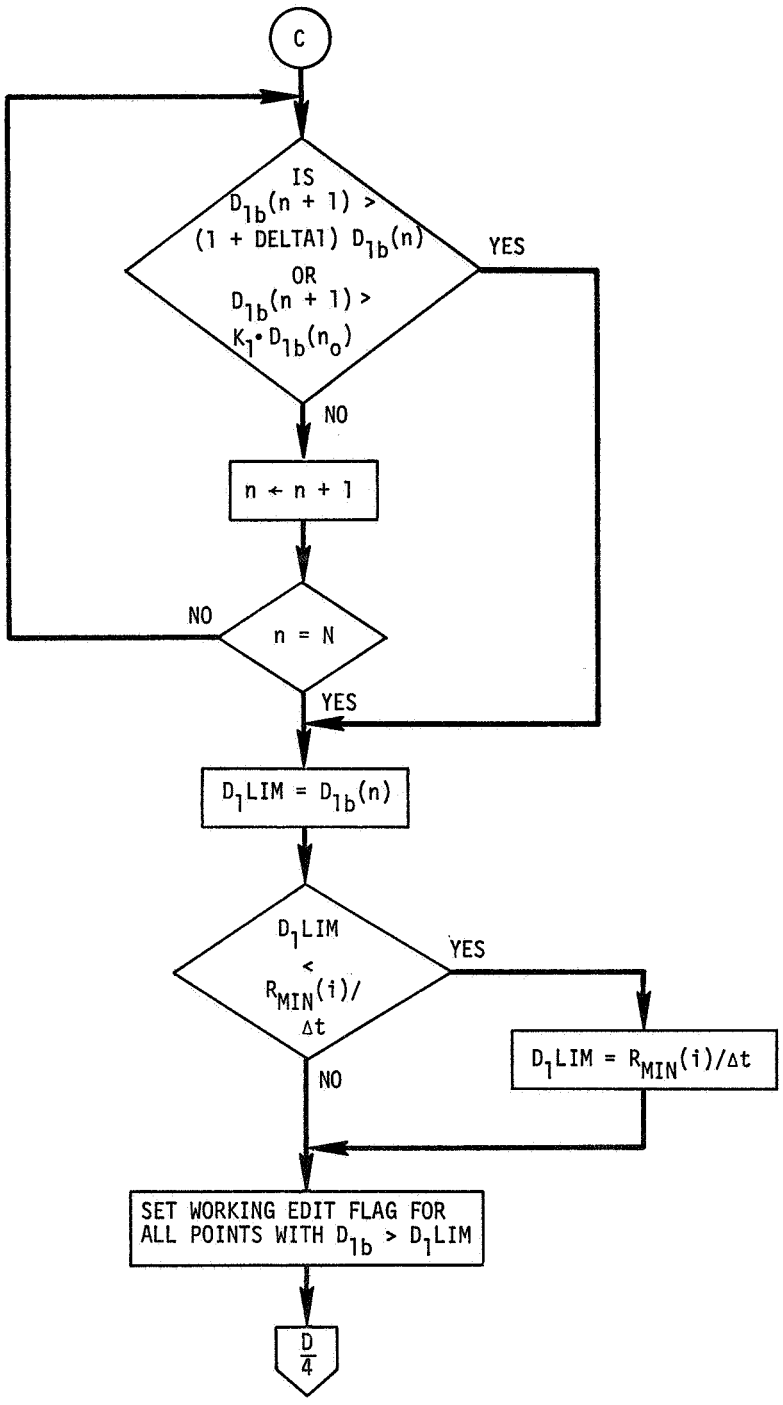
APPENDIX A

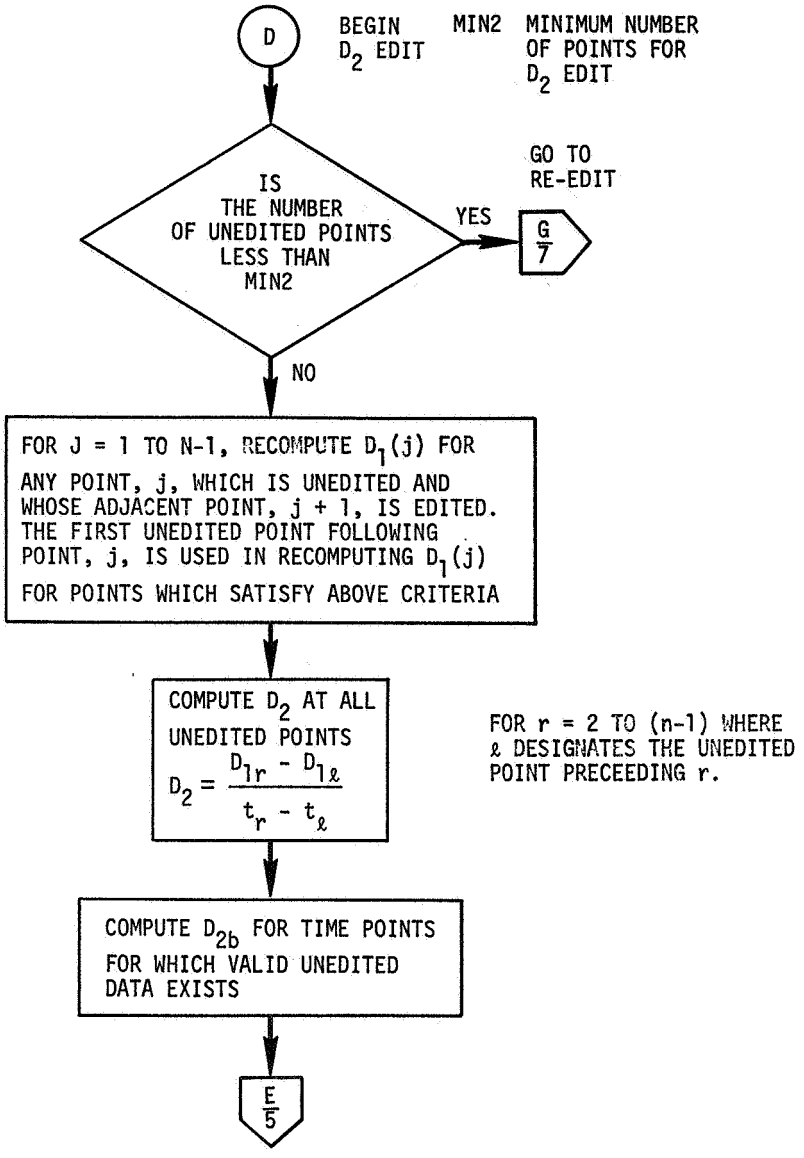
FLOW CHART FOR THE DATA EDITING PROCESSOR

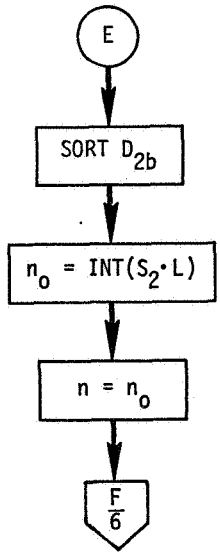

ENTER EDITOR
INPUTS: R_{OB}, DT, NF
OUTPUTS: P_{ed}, EDIT STOP FLAG, EDIT ALTER FLAG, BATCH EDITED FLAG
CLEAR EDIT ALTER FLAG
IS THE NUMBER OF FRAMES LESS THAN MIN1
YES
J/8
NO
CLEAR WORKING EDIT FLAGS
A
SET I TO INDEX FOR CURRENT MEASUREMENT TYPE
MIN1 MINIMUM NUMBER OF POINTS FOR D_1 EDIT
ARE THERE LESS THAN MIN1 VALID POINTS OF THIS MEASUREMENT TYPE
YES
I/9
NO
SET WORKING EDIT FLAGS FOR ALL POINTS WITH RESIDUALS GREATER THAN R_{MAX}
IS THE FRACTION OF POINTS EDITED GREATER THAN ELIM
YES
J/8
NO
B/2

Page 1 of 9

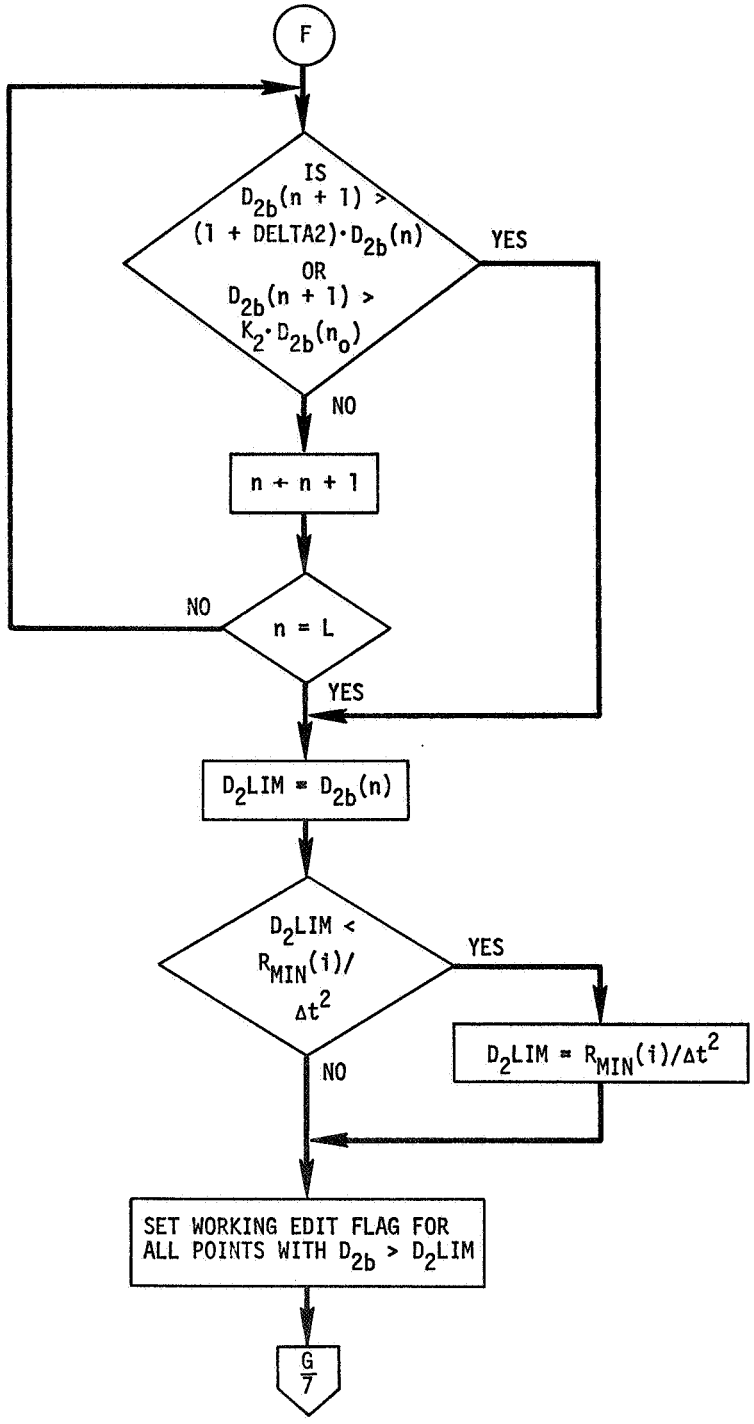

B
ARE THERE LESS THAN MIN1 POINTS OF THIS DATA TYPE WITH WORKING EDIT FLAGS CLEAR
YES
J/8
NO
CALCULATE D_1 FOR ALL TIME POINTS FOR WHICH VALID UNEDITED DATA EXISTS
CALCULATE D_{1b} FOR ALL TIME POINTS FOR WHICH VALID UNEDITED DATA EXISTS
DETERMINE Δt FROM DATA
SORT D_{1b} IN ASCENDING ORDER
$n_o = INT(S_1 \cdot N)$
$n = n_o$
C/3
Δt = NORMAL TIME BETWEEN POINTS
$\Delta t = \frac{t_L - t_F}{(no.\ frames - 1)}$
t_L = LAST TIME
t_F = FIRST TIME
N = NUMBER OF VALID UNEDITED POINTS
n_o = STARTING POINT FOR EDIT LIMIT SEARCH
INT = INTEGER VALUE FUNCTION

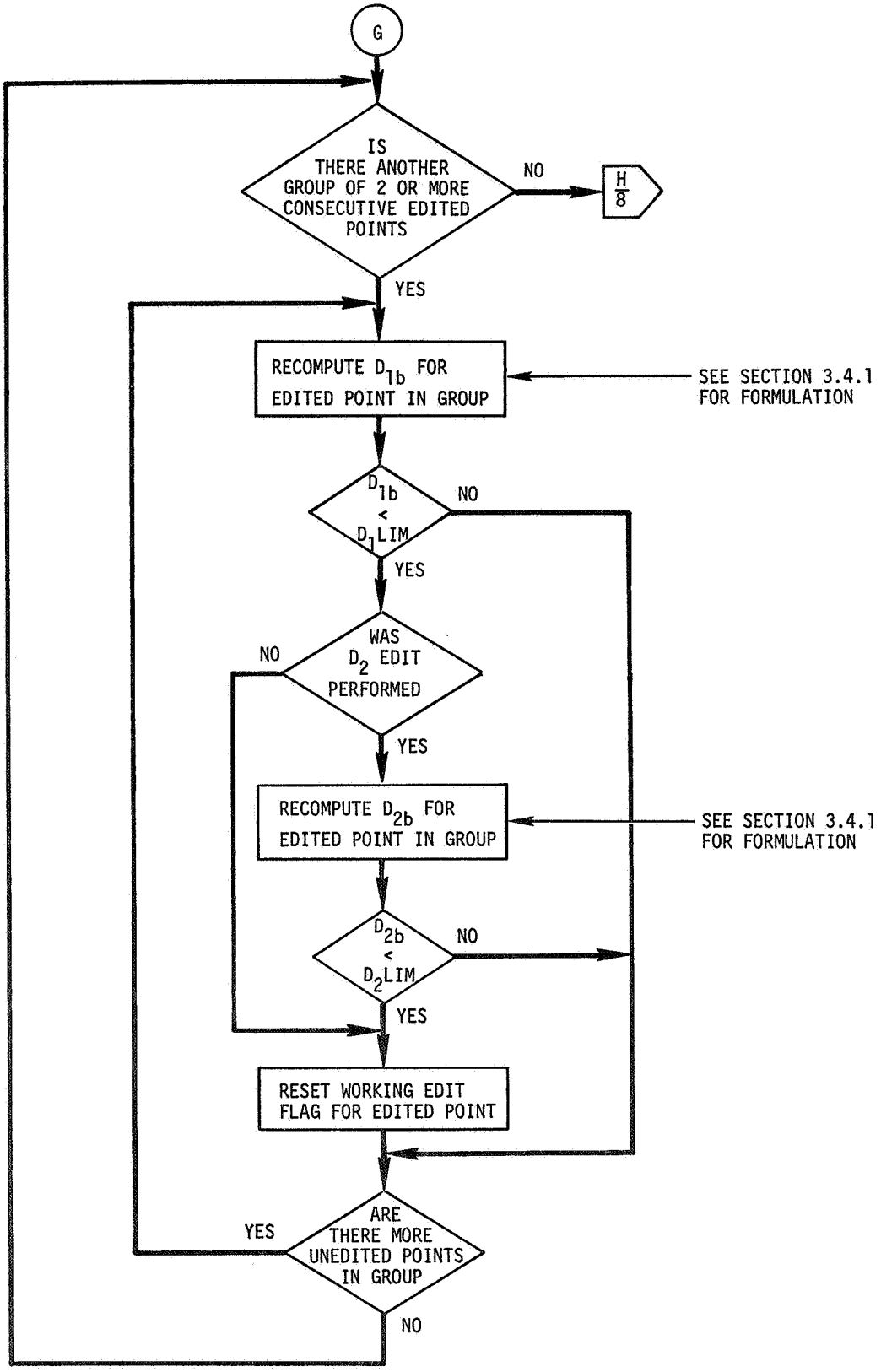

C
IS $D_{1b}(n+1) > (1 + DELTA1)\ D_{1b}(n)$ OR $D_{1b}(n+1) > K_1 \cdot D_{1b}(n_o)$
YES
NO
$n \leftarrow n + 1$
NO
$n = N$
YES
$D_1LIM = D_{1b}(n)$
$D_1LIM < R_{MIN}(i)/\Delta t$
YES
$D_1LIM = R_{MIN}(i)/\Delta t$
NO
SET WORKING EDIT FLAG FOR ALL POINTS WITH $D_{1b} > D_1LIM$
D/4

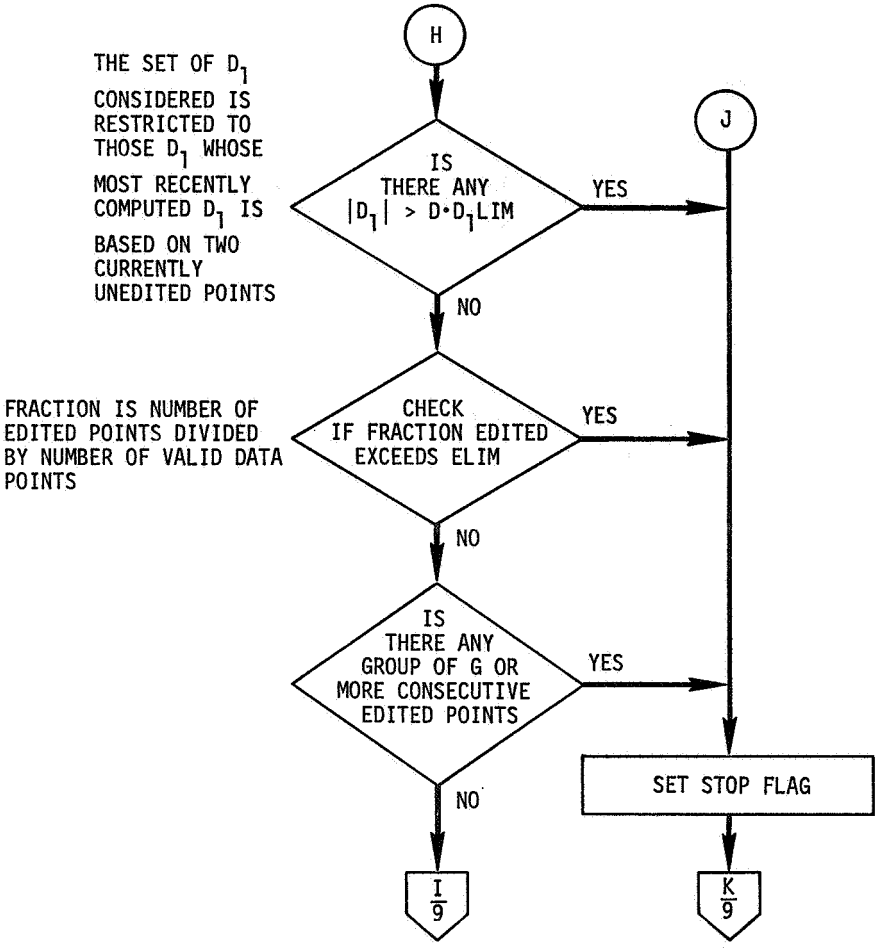

D
BEGIN D_2 EDIT
MIN2 MINIMUM NUMBER OF POINTS FOR D_2 EDIT
IS THE NUMBER OF UNEDITED POINTS LESS THAN MIN2
YES
GO TO RE-EDIT
G/7
NO
FOR J = 1 TO N-1, RECOMPUTE $D_1(j)$ FOR ANY POINT, j, WHICH IS UNEDITED AND WHOSE ADJACENT POINT, j + 1, IS EDITED. THE FIRST UNEDITED POINT FOLLOWING POINT, j, IS USED IN RECOMPUTING $D_1(j)$ FOR POINTS WHICH SATISFY ABOVE CRITERIA
COMPUTE D_2 AT ALL UNEDITED POINTS
$D_2 = \frac{D_{1r} - D_{1\ell}}{t_r - t_\ell}$
FOR r = 2 TO (n-1) WHERE ℓ DESIGNATES THE UNEDITED POINT PRECEEDING r.
COMPUTE D_{2b} FOR TIME POINTS FOR WHICH VALID UNEDITED DATA EXISTS
E/5

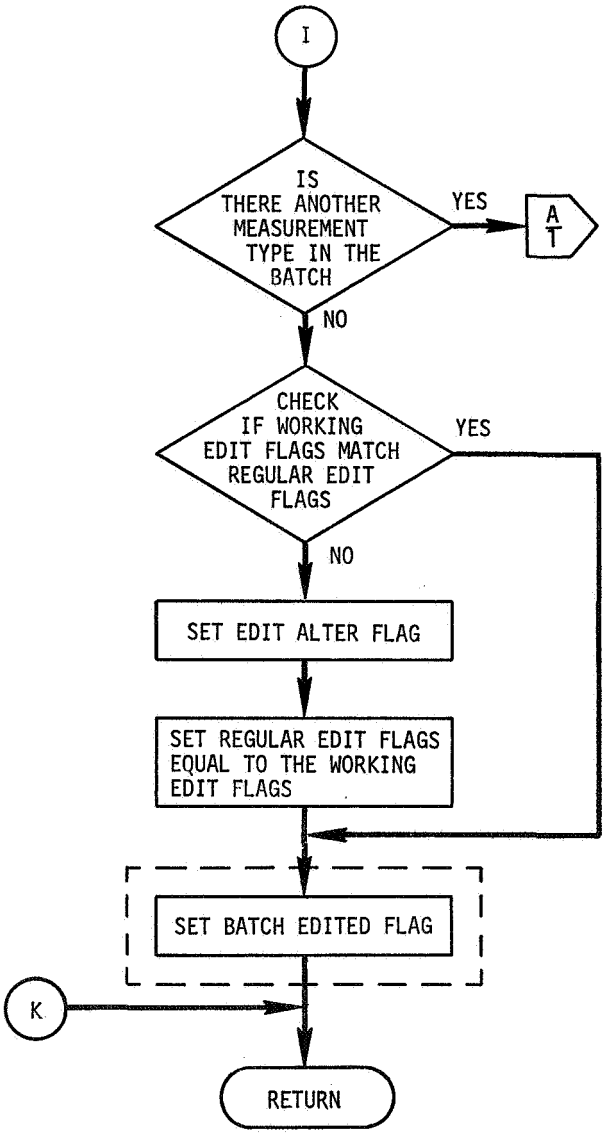


L = NUMBER OF VALID UNEDITED POINTS


F
IS
$D_{2b}(n+1) >$
$(1 + DELTA2) \cdot D_{2b}(n)$
OR
$D_{2b}(n+1) >$
$K_2 \cdot D_{2b}(n_o)$
YES
NO
$n \leftarrow n + 1$
NO
$n = L$
YES
$D_2LIM = D_{2b}(n)$
$D_2LIM <$
$R_{MIN}(i)/$
Δt^2
YES
$D_2LIM = R_{MIN}(i)/\Delta t^2$
NO
SET WORKING EDIT FLAG FOR
ALL POINTS WITH $D_{2b} > D_2LIM$
G
7

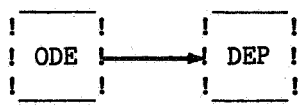

G
IS THERE ANOTHER GROUP OF 2 OR MORE CONSECUTIVE EDITED POINTS
NO
H/8
YES
RECOMPUTE D_{1b} FOR EDITED POINT IN GROUP
SEE SECTION 3.4.1 FOR FORMULATION
D_{1b} < D_1LIM
NO
YES
WAS D_2 EDIT PERFORMED
NO
YES
RECOMPUTE D_{2b} FOR EDITED POINT IN GROUP
SEE SECTION 3.4.1 FOR FORMULATION
D_{2b} < D_2LIM
NO
YES
RESET WORKING EDIT FLAG FOR EDITED POINT
ARE THERE MORE UNEDITED POINTS IN GROUP
YES
NO


H
THE SET OF D_1 CONSIDERED IS RESTRICTED TO THOSE D_1 WHOSE MOST RECENTLY COMPUTED D_1 IS BASED ON TWO CURRENTLY UNEDITED POINTS
IS THERE ANY $|D_1| > D \cdot D_1LIM$
YES
NO
J
FRACTION IS NUMBER OF EDITED POINTS DIVIDED BY NUMBER OF VALID DATA POINTS
CHECK IF FRACTION EDITED EXCEEDS ELIM
YES
NO
IS THERE ANY GROUP OF G OR MORE CONSECUTIVE EDITED POINTS
YES
NO
SET STOP FLAG
I/9
K/9


I
IS THERE ANOTHER MEASUREMENT TYPE IN THE BATCH
YES
A T
NO
CHECK IF WORKING EDIT FLAGS MATCH REGULAR EDIT FLAGS
YES
NO
SET EDIT ALTER FLAG
SET REGULAR EDIT FLAGS EQUAL TO THE WORKING EDIT FLAGS
SET BATCH EDITED FLAG
K
RETURN

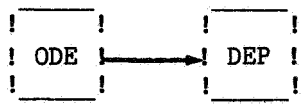
APPENDIX B

INTERFACE TABLES



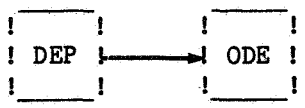
INTERFACE TABLE

ODE parameter	DEP parameter	Unit	Description
	D	--	Limit that sets the size of the discontinuity that can be accepted (D>1).
	DELTA1	--	User-definable constant greater than 0 which sets the smallest fractional increase between D_{1b} values that will trigger the automatic edit level setting logic.
	DELTA2	--	User-definable constant greater than 0 which sets the smallest fractional increase between D_{2b} values that will trigger the automatic edit level setting logic.
	ELIM	--	Limit that sets the fraction of the data that that can be edited before the edit stop flag is set (0<ELIM<1).
	G	--	Integer limit that determines how many consecutively edited points can be accepted (G>1).
	DT	--	Indicates the measurement types contained within the current batch.
	K1	--	User-definable constant greater than 1 which sets the largest overall increase in D_{1b} values relative to their reference that will be accepted before the automatic level setting logic is triggered.
	K2	--	User-definable constant greater than 1 which sets the largest overall increase in D_{2b} values relative to their reference that will be accepted before the automatic edit limit setting logic is triggered.
	MIN1	--	User-definable integer constant that sets the lower limit to the number of points for which a D_1 edit will be performed (MIN1>2).



INTERFACE TABLE

ODE parameter	DEP parameter	Unit	Description
	MIN2	--	User-definable integer constant that sets the lower limit to the number of points for which a D_2 edit will be performed (MIN2>3).
	NF	--	Number of frames in batch.
	R_{0B}	Internal	Batch of observation residuals
	RMIN(I)	Internal	Set of user-definable constants, one for each measurement type, that sets a lower bound below which editing will not occur.
	RMAX(I)	Internal	Set of user-definable constants, one for each measurement type, that sets an upper bound beyond which preediting will always occur.
	S1	--	User-definable limit between 0 and 1.0 which sets the fraction D_{1b} values that will not be edited by controlling the reference of the automatic edit limit setting logic.
	S2	--	User-definable limit between 0 and 1.0 which sets the fraction of D_{2b} values that will not be edited by controlling the reference of the automatic edit limit logic.
T	T_r	Hours	Time tags associated with each residual in a batch.



INTERFACE TABLE

DEP parameter	ODE parameter	Unit	Description
Edit stop flag	Edit stop flag	Flag	Indicates that the editor has encountered a situation calling for user examination.
Edit alter flag	Edit alter flag	Flag	Indicates that the current pass through the editor has caused a change in the batch edit status.
Edit flags	P_{ed}	Flags	Indicates which points in batch of observations were edited.
Batch edited flag	Batch edited flag	Flag	Indicates that the batch has been edited.
Number of edited points			Indicates the number of points edited for each measurement type in the batch.

U.S. GOVERNMENT PRINTING OFFICE: 1979—671-014/22 52

DISTRIBUTION FOR JSC IN 78-FM-30, VOL. VIII

JM6/Technical Library (2)
JM61/Center Data Management (3)
CF3/J. Greene
FM8/E. Schiesser
J. Williamson (5)
J. Currie
W. York
J. Weaver
W. Wollenhaupt
R. Osburn
FM14/Report Control (20)
A. Wiseman
B. Woodland
FM17/L. Hartley
FS5/M. Dixon
J. Mendiola
FS15/R. Brown (2)
IBM/H. Norman
R. Rich
W. Goodyear
F. Riddle
C. Waund
A. Stevenson (8)
MDTSCO/T. Rich
R. Theis
TRW/O. Bergman (15)